

Run Video Chat within your Unity application (Windows)

Being able to communicate with across mobile platforms is great but when you add the ability to connect to the Windows platform you begin to open a whole new world of possibilities. With our Agora Video SDK for Unity3D you can add cross-platform functionality to Windows, Mac, Android, and iOS. Follow along and I will show you how to get video communication up and running in no time on your Windows device.

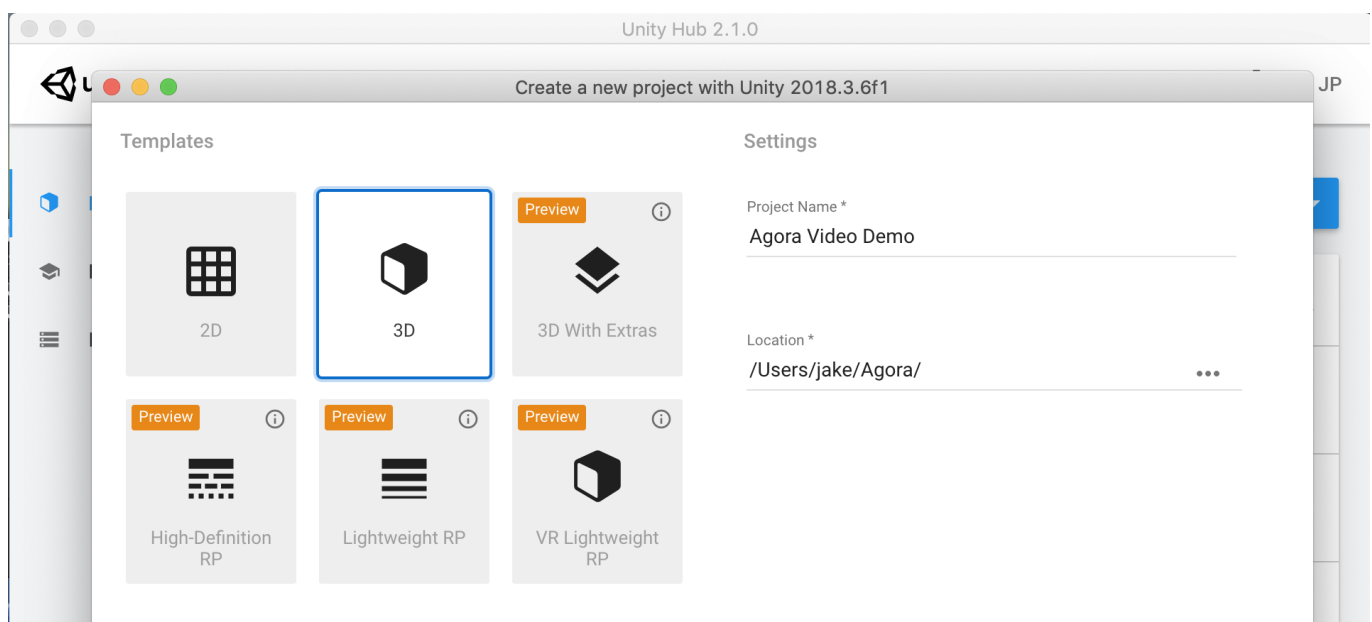
Getting a sample Windows app up and running is simple and took less than 10 minutes. Let's run through the steps together.

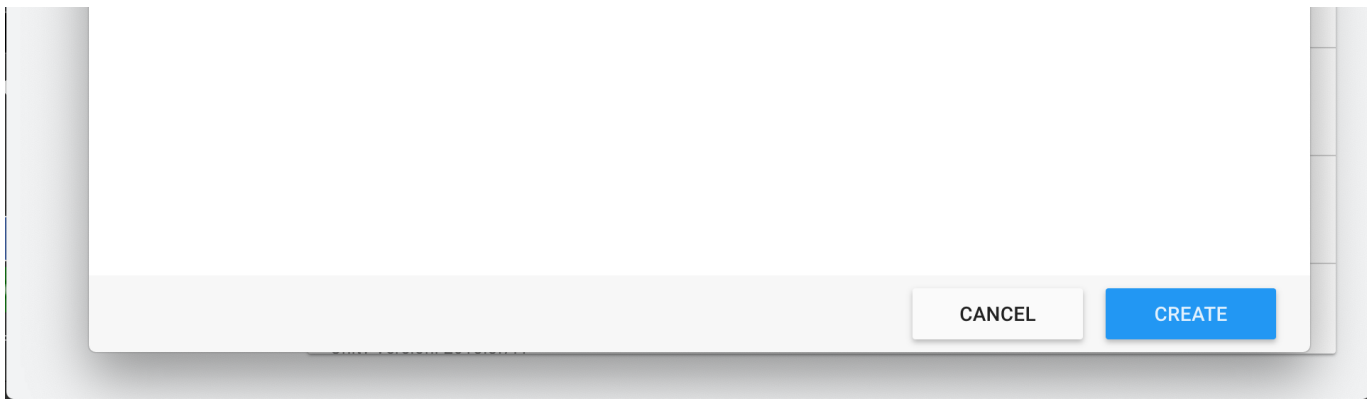
Prerequisites

- Unity Editor
- A developer account with Agora.io

Getting Started

To start, open Unity and create a blank new project named **Agora Video Demo**.

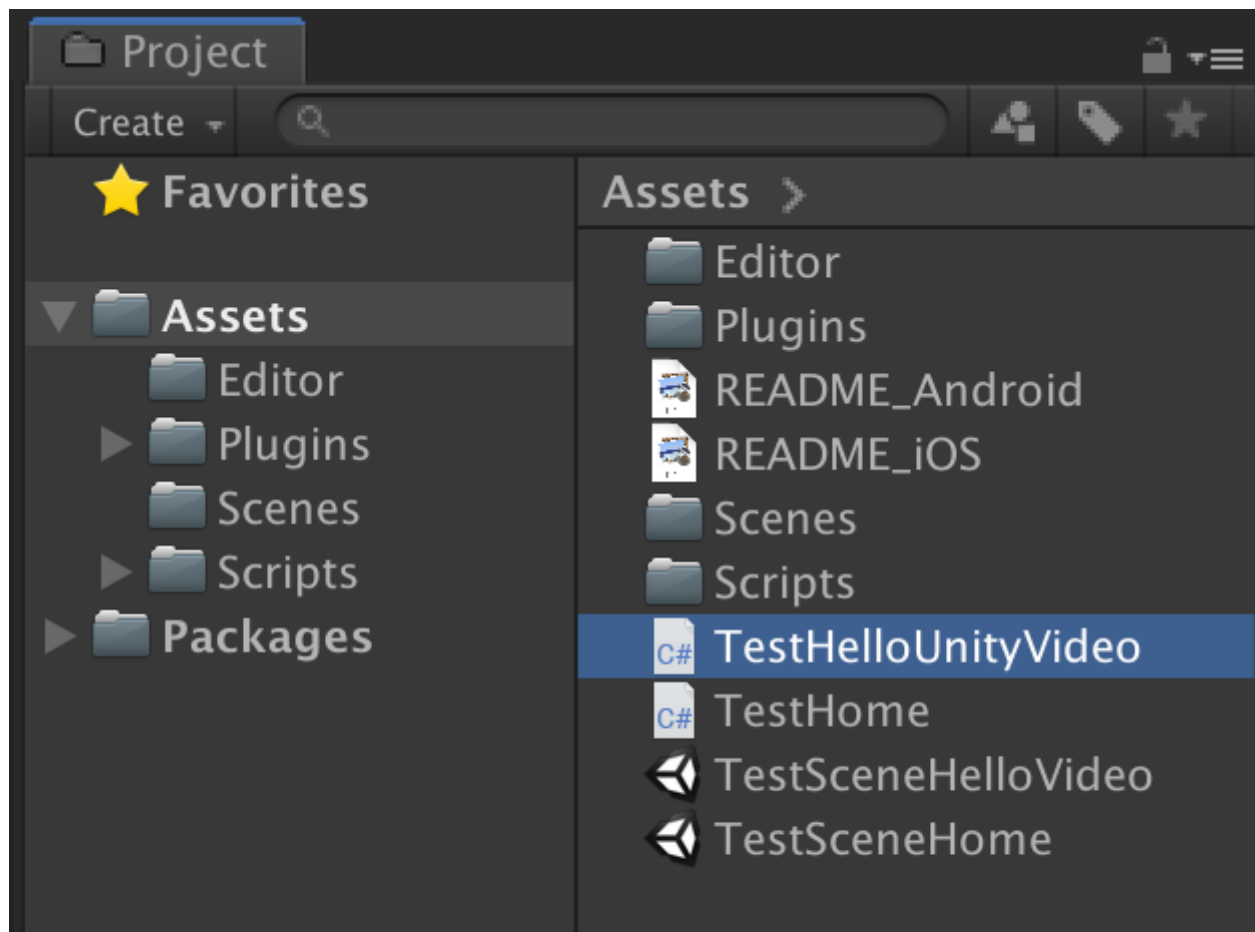




The next step is to navigate to the Unity Store (*if you are in the scene view, click the Unity Store tab*) and search for “Agora Video SDK”. Once the plugin page has loaded, go ahead and click **Download**. Once the plugin page has loaded, go ahead and click **Download**. Once the download is complete, click and **Import** the assets into your project.

Update code

Great! Now that we have the plugin downloaded and imported to our project we need to make 1 quick update before we can build and run our app. Double click the `TestHelloUnityVideo.cs` file within the assets panel to edit it.





The file will open in Visual Studio (*Unity's default code editor*). Let's scroll down to line 16 .

```
12     public class TestHelloUnityVideo : MonoBehaviour {  
13  
14         // PLEASE KEEP THIS App ID IN SAFE PLACE  
15         // Get your own App ID at https://dashboard.agora.io/  
16         private static string appId = "YOUR_APPID";  
17         // load agora engine
```

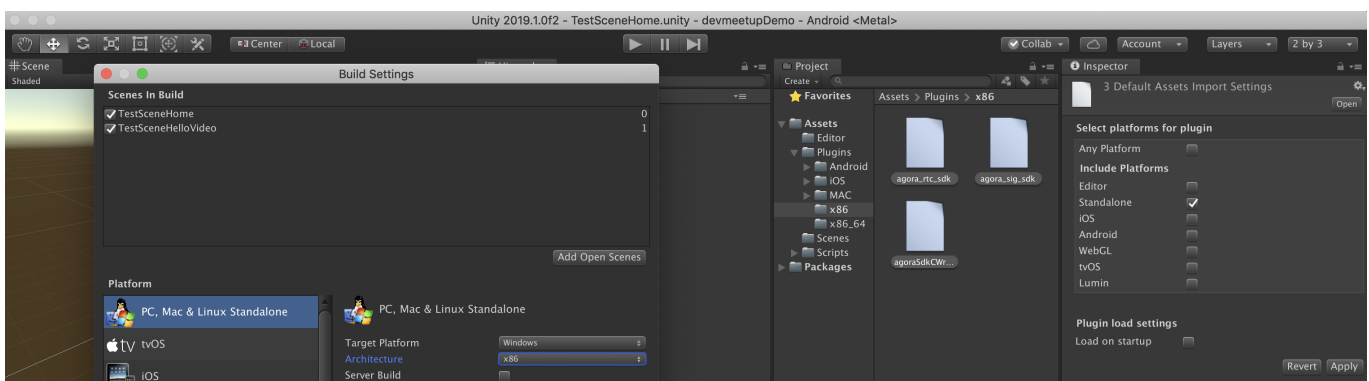
You will need to change the App ID to a new one you create on the Agora.io website. Don't worry about charges during early development, they give you the first 10,000 minutes **free** every month

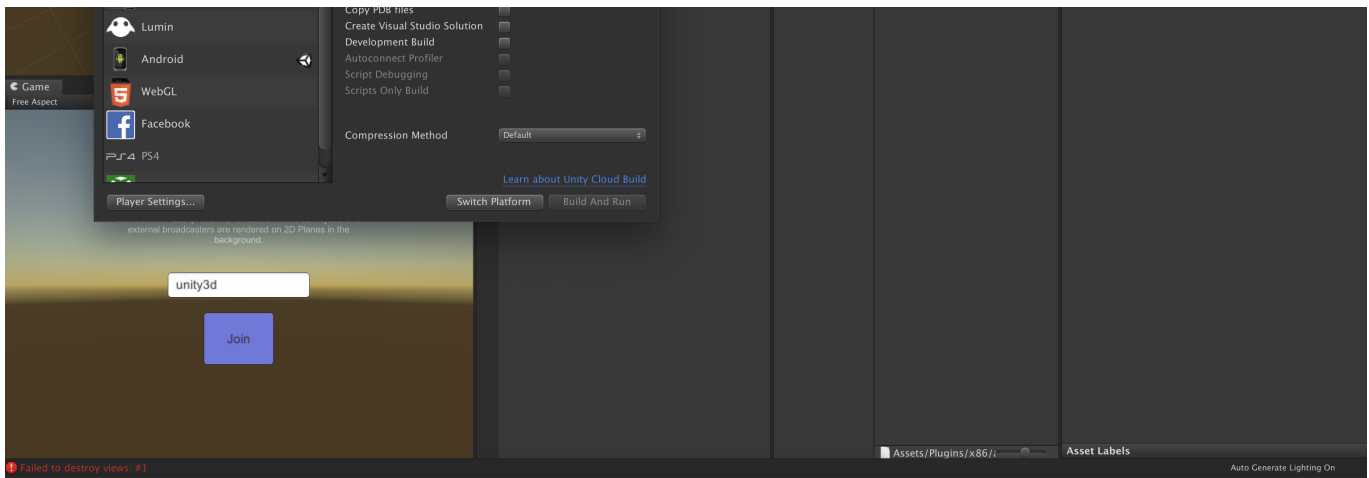
At this point, we need to log into our Agora.io developer account and create a new AppID or select an existing one. Then replace "YOUR_APPID " on line 16 by copying and pasting over it. Make sure to save your changes.

We can scroll through the TestHelloUnityVideo.cs file to see all the various event callbacks the Agora.io Voice SDK for Unity offers.

Set Plugin Identities

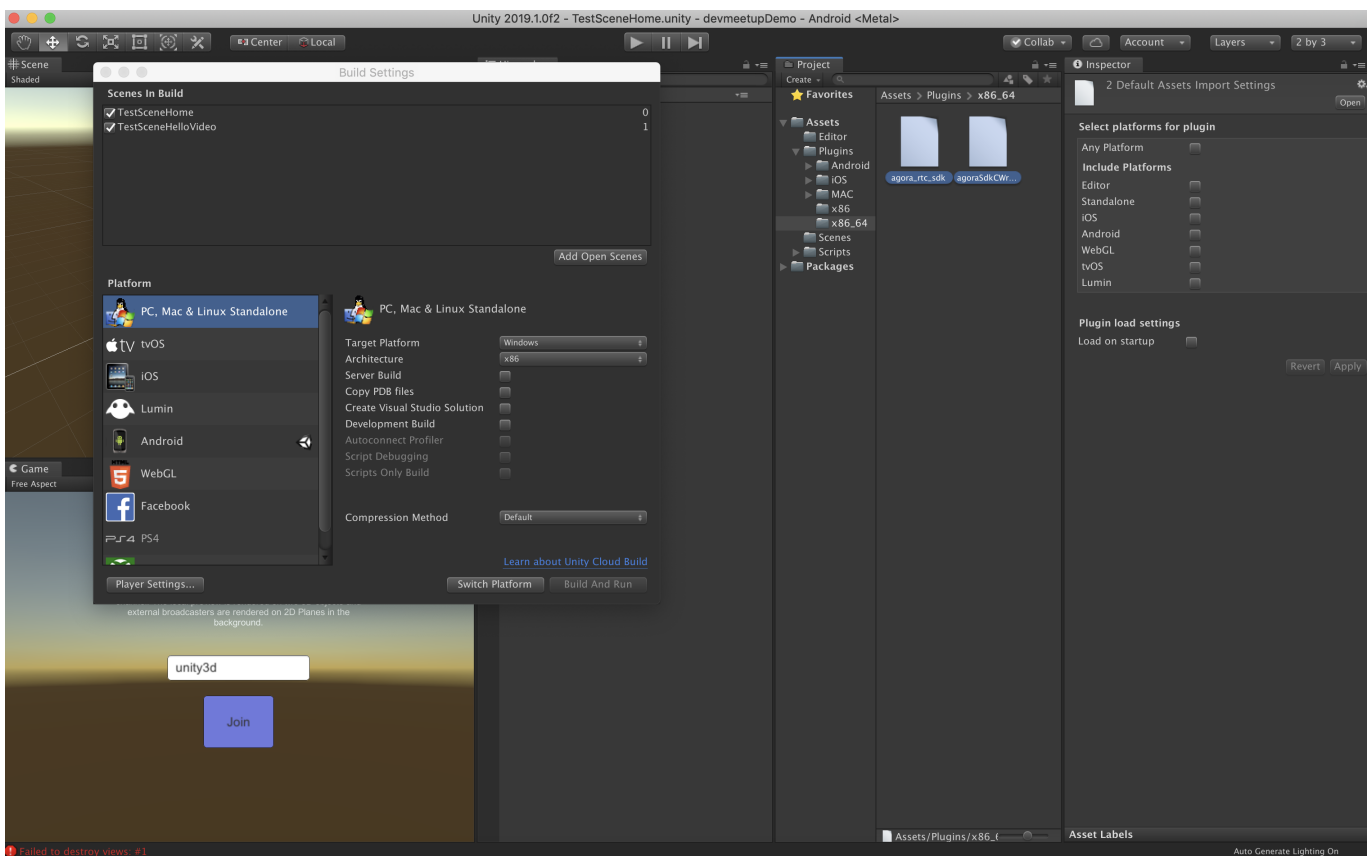
Because the Agora.io Video chat app has cross-functionality with Android, iOS, Mac, and Windows we will set the plugin identities. You will need to also choose between x86 and x86_64 and set the identities accordingly. Here I will go into Assets>Plugins>x86 folder and select only Standalone so I can run it on my Windows device and then hit Apply.





Here I choose x86 as my Architecture

So that the plugins don't collide we will then need to go into Assets>Plugins>x86_64 and select all the files. Then go into the Inspector and uncheck everything and hit apply.

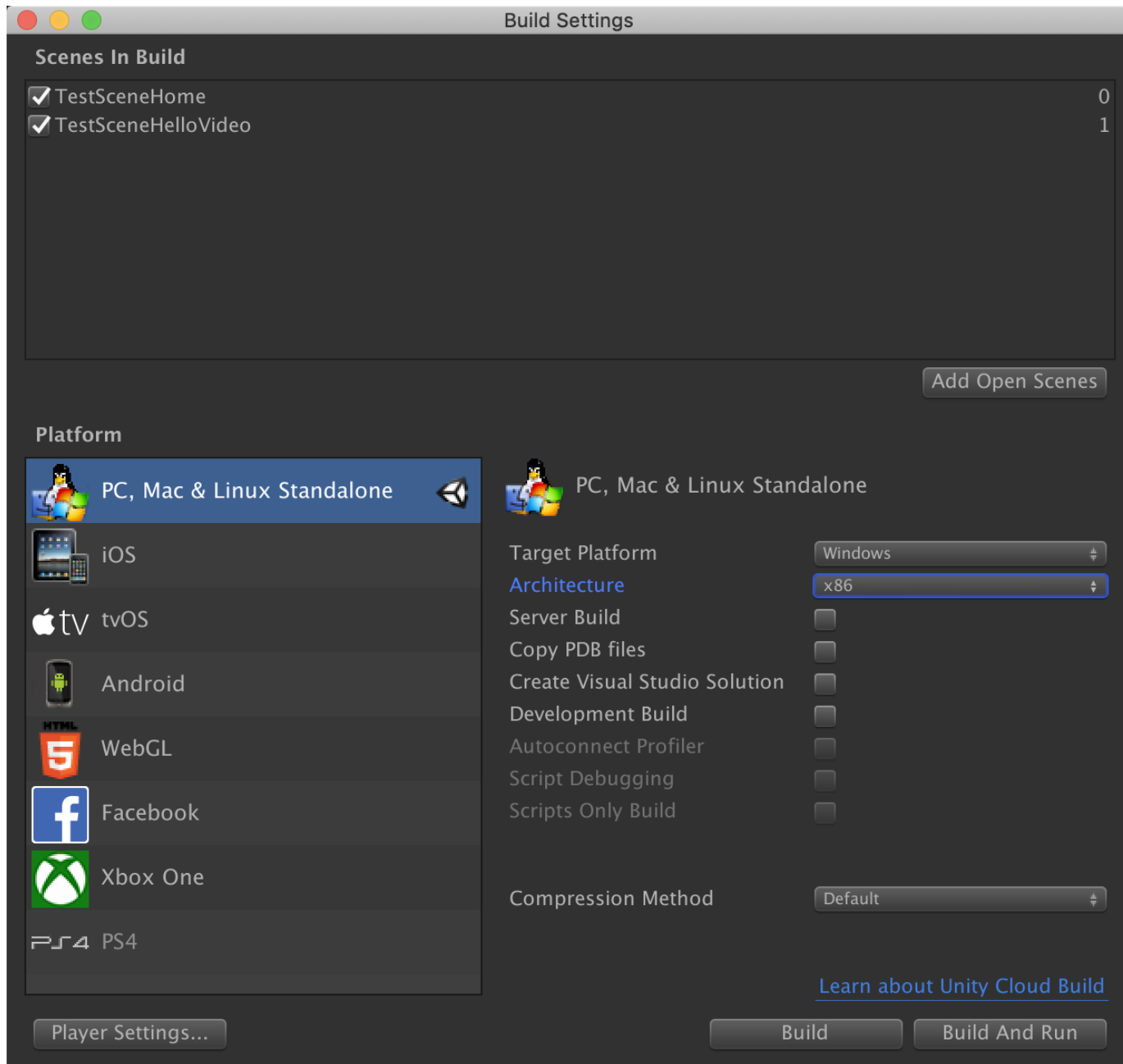


Deselecting the x86_64 plugin will keep you from getting a collision error.

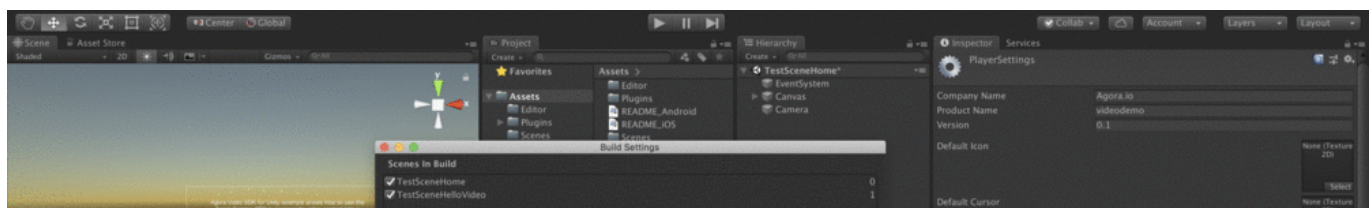
Running the Sample Application

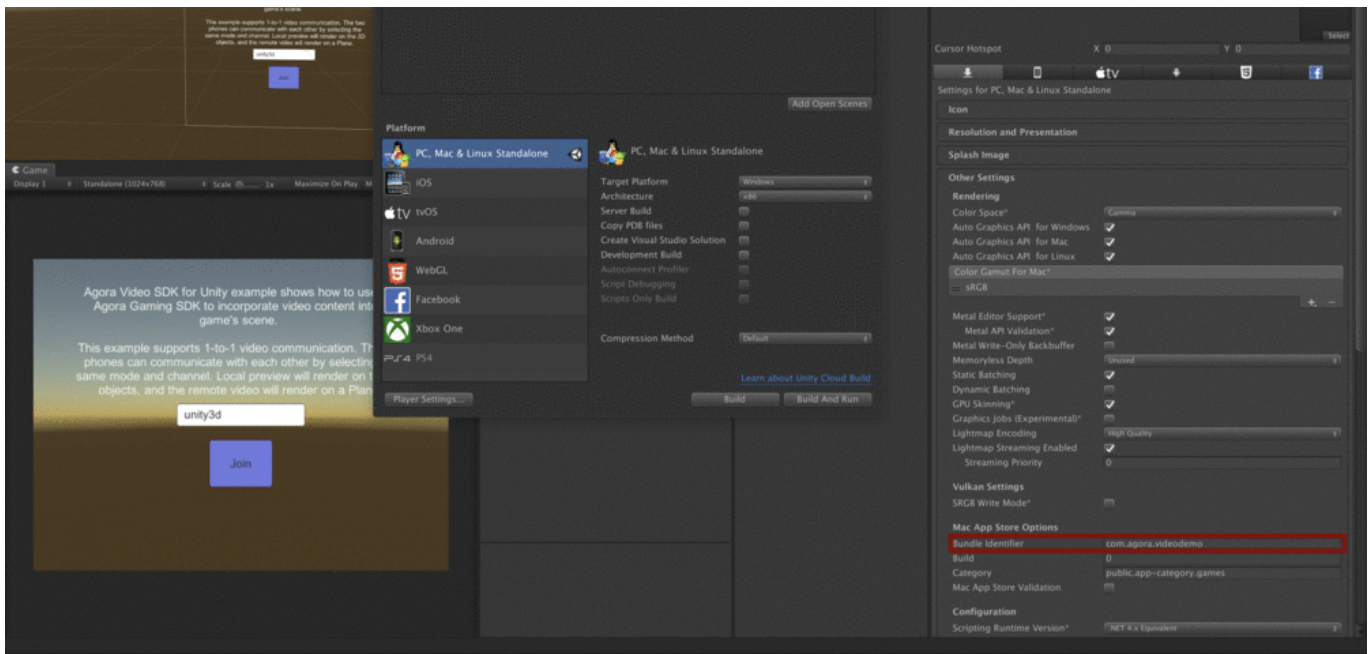
Open the **Build Settings** and drag the `TestSceneHome.unity` & `TestSceneHelloVideo.unity` scenes from the assets list into the “*Scenes in Build*” list. If it is

not already selected, select PC, Mac & Linux from the *Platform* list and select *Windows* as the **Target Platform**. Since the Video SDK 2.9.0 update, x86 and x86_64 architectures are both supported. If necessary click *Switch Platform*.



Once Unity finishes its setup process, open the *Player Settings*. We will need to make sure our Windows app also has a unique package name, I chose `com.agora.videodemo`.





Then click **Build And Run**. Unity will prompt you to save your project, I create a folder for my builds named Builds and Create a folder inside of that named WindowsBuilds to hold my Windows builds, this will keep you from having issues in the future if you want to use the same project to run on Mac, iOS, or Android.

Other Resources

- The complete API documentation is available in the Document Center.
- For technical support, submit a ticket using the Agora Dashboard or reach out directly to our Developer Relations team devrel@agora.io